

09/28/00

A

**SKJERVEN**  
**MORRILL**  
**MACPHERSON LLP**

Docket No.: M-8639 US

September 26, 2000

Box Patent Application  
 Commissioner For Patents  
 Washington, D. C. 20231

Enclosed herewith for filing is a patent application, as follows:

Inventors: Hsu, Ivy P.S.; Cheung, David C.Y.; Jalan, Rajkumar R.

Title: Global Server Load Balancing

X Return Receipt Postcard  
X This Transmittal Letter (in duplicate)  
13 Page Specification (not including claims)  
6 pages of Claims  
1 Page Abstract  
3 Sheets of Drawings (Figures 1, 2, and 3)  
3 Page Declaration For Patent Application and Power of Attorney  
1 Page Recordation Form Cover Sheet (in duplicate)  
2 Page Assignment

**CLAIMS AS FILED**

For	Number			Number		Rate		Basic Fee
	<u>Filed</u>			<u>Extra</u>				
Total Claims	26	-20	=	6	x	\$18.00	=	\$ 108.00
Independent Claims	2	-3	=	0	x	\$78.00	=	\$ 0.00
<input type="checkbox"/>	Fee of _____ for the first filing of one or more multiple dependent claims per application							\$
<input type="checkbox"/>	Fee for Request for Extension of Time							\$

Please make the following charges to Deposit Account 19-2386:

- ☒ Total fee for filing the patent application in the amount of \$ 798.00  
☒ The Commissioner is hereby authorized to charge any additional fees, which may be required, or credit any overpayment to Deposit Account 19-2386.

**EXPRESS MAIL LABEL NO:****EL 375 516 184 US**

Respectfully submitted,



Edward C. Kwok  
 Attorney for Applicants  
 Reg. No. 33,938

GLOBAL SERVER LOAD BALANCING

Ivy Pei-Shan Hsu

David Chun Ying Cheung

5

Rajkumar Ramniranjan Jalan

Background of the Invention

1. Field of the Invention

10 The present invention relates to load balancing  
among servers. More particularly, the present  
invention relates to achieving load balancing by, in  
response to resolving a DNS query by a client,  
providing the address of a server that is expected to  
serve the client with a high performance in a given  
15 application.

2. Description of the Related Art

Under the TCP/IP protocol, when a client provides  
a symbolic name ("URL") to request access to an  
application program or another type of resource, the  
20 host name portion of the URL needs to be resolved into  
an IP address of a server for that application program  
or resource. For example, the URL (e.g.,  
http://www.foundrynet.com/index.htm) includes a host  
name portion www.foundrynet.com that needs to be  
25 resolved into an IP address. The host name portion is  
first provided by the client to a local name resolver,  
which then queries a local DNS server to obtain a  
corresponding IP address. If a corresponding IP  
address is not locally cached at the time of the query,  
30 or if the "time-to-live" (TTL) of a corresponding IP  
address cached locally has expired, the DNS server then  
acts as a resolver and dispatches a recursive query to  
another DNS server. This process is repeated until an

authoritative DNS server for the domain (i.e. foundrynet.com, in this example) is reached. The authoritative DNS server returns one or more IP addresses, each corresponding to an address at which a server hosting the application ("host server") under the host name can be reached. These IP addresses are propagated back via the local DNS server to the original resolver. The application at the client then uses one of the IP addresses to establish a TCP connection with the corresponding host server. Each DNS server caches the list of IP addresses received from the authoritative DNS for responding to future queries regarding the same host name, until the TTL of the IP addresses expires.

To provide some load sharing among the host servers, many authoritative DNS servers use a simple round-robin algorithm to rotate the IP addresses in a list of responsive IP addresses, so as to distribute equally the requests for access among the host servers.

The conventional method described above for resolving a host name to its IP addresses has several shortcomings. First, the authoritative DNS does not detect a server that is down. Consequently, the authoritative DNS server continues to return a disabled host server's IP address until an external agent updates the authoritative DNS server's resource records. Second, when providing its list of IP addresses, the authoritative DNS sever does not take into consideration the host servers' locations relative to the client. The geographical distance between the server and a client is a factor affecting the response time for the client's access to the host server. For example, traffic conditions being equal, a client from Japan could receive better response time from a host server in Japan than from a host server in New York.

Further, the conventional DNS algorithm allows invalid IP addresses (e.g., that corresponding to a downed server) to persist in a local DNS server until the TTL for the invalid IP address expires.

5

### Summary of the Invention

The present invention provides an improved method and system for serving IP addresses to a client, based on a selected set of performance metrics. In accordance with this invention, a global server load-balancing (GSLB) switch is provided as a proxy for an authoritative DNS server, together with one or more site switches each associated with one or more host servers. Both the GSLB switch and the site switch can be implemented using the same type of switch hardware. Each site switch provides the GSLB switch with current site-specific information regarding the host servers associated with the site switch. Under the present invention, when an authoritative DNS server resolves a host name in a query and returns one or more IP addresses, the GSLB switch filters the IP addresses using the performance metrics compiled from the site-specific information collected from the site switches. The GSLB switch then returns a ranked or weighted list of IP addresses to the inquirer. In one embodiment, the IP address that is estimated to provide the best expected performance for the client is placed at the top of the list. Examples of suitable performance metrics include availability metrics (e.g., a server's or an application's health), load metrics (e.g., a site switch's session capacity or a corresponding preset threshold), and proximity metrics (e.g., a round-trip time between the site switch and a requesting DNS server, the geographic location of the host server, the topological distance between the host server and the

client program). (A topological distance is the number of hops between the server and the client). Another proximity metrics is the site switch's "flashback" speed (i.e., how quickly a switch receives a health check result). The ordered list can also be governed by other policies, such as the least selected host server.

The present invention is better understood upon consideration of the detailed description of the preferred embodiments below, in conjunction with the accompanying drawings.

#### Brief Description of the Drawings

Fig. 1 illustrates a global server load balancing configuration.

Fig. 2 illustrates in a flow chart one embodiment of the algorithm for selecting the "best" address from the list of addresses supplied by the authoritative DNS.

Fig. 3 is a block diagram showing the functional modules of GSLB switch 12 and site switch 18a relevant to the global server load balancing function.

#### Detailed Description of the Preferred Embodiments

Fig. 1 illustrates one embodiment of the present invention that provides a global server load balancing configuration. As shown in Fig. 1, global server load balancing (GSLB) switch 12 is connected to Internet 14 and acts as a proxy to an authoritative Domain Name System (DNS) server 16 for the domain "foundrynet.com" (for example). That is, while the actual DNS service is provided by DNS server 16, the IP address known to the rest of the Internet for the authoritative DNS



detail below. In the remainder of this detailed description, for the purpose of illustrating the present invention only, the list of IP addresses returned are assumed to be the virtual IP addresses  
5 configured on the proxy servers at switches 18A, 18B, 22A and 22B (sites 20 and 24). In one embodiment, GSLB switch 12 determines which site switch would provide the best expected performance (e.g., response time) for client 28 and returns the IP address list with a  
10 virtual IP address configured at that site switch placed at the top. (Within the scope of the present invention, other forms of ranking or weighting the IP addresses in the list can also be possible.) Client program 28 can receive the ordered list of IP  
15 addresses, and typically selects the first IP address on the list to access the corresponding host server.

Fig. 3 is a block diagram showing the functional modules of GSLB switch 12 and site switch 18a relevant to the global server load balancing function. As shown  
20 in Fig. 3, GSLB 12 includes a GSLB switch controller 401, health check module 402, DNS proxy module 403, metric agent 404, routing metric collector 405, and site-specific metric collector 406. GSLB switch controller 401 provides general control functions for  
25 the operation of GSLB switch 12. Health check module 402 is responsible for querying, either periodically or on demand, host servers and relevant applications hosted on the host servers to determine the "health" (i.e., whether or not it is available) of each host  
30 server and each relevant application. Site-specific metric collector 406 communicates with metric agents in site-specific switches (e.g., Fig. 3 shows site-specific metric collector 406 communicating with site-specific metric agent 407) to collect site-specific  
35 metrics (e.g., number of available sessions on a

specific host server). Similarly, routing metric collector 405 collects routing information from routers (e.g., topological distances between nodes on the Internet). Fig. 3 shows, for example, router 408 providing routing metric collector 405 with routing metrics (e.g., topological distance between the load balancing switch and the router), using the Border Gateway Protocol (BGP). DNS proxy module 403 (a) receives incoming DNS requests, (b) provides the host names to be resolved to DNS server 16, (c) receives from DNS server 16 a list of responsive IP addresses, (d) orders the IP addresses on the list received from DNS server 16 according to the present invention, using the metrics collected by routing-metric collector 405 and site specific collector 406, and values of any other relevant parameter, and (e) provides the ordered list of IP addresses to the requesting DNS server. Since GSLB switch 12 can also act as a site switch, GSLB switch 12 is provided site-specific metric agent 404 for collecting metrics for a site-specific metric collector.

In one embodiment, the metrics used in a GSLB switch includes (a) the health of each host server and selected applications, (b) each site switch's session capacity threshold, (c) the round trip time (RTT) between a site switch and a client in a previous access, (d) the geographical location of a host server, (e) the current available session capacity in each site switch, (f) the "flashback" speed between each site switch and the GSLB switch (i.e., how quickly each site switch responds to a health check from the GSLB switch), and (g) a policy called the "Least Response selection" (LRS) which prefers the site least selected previously. Many of these performance metrics can be provided default values. Each individual metric can be

used in any order and each metric can be disabled. In one embodiment, the LRS metric is always enabled.

Figure 2 illustrates in a flow diagram one embodiment of an optimization algorithm utilized by  
5 GSLB switch 12 to process the IP address list received from DNS server 16, in response to a query resulting from client program 28. As shown in Fig. 2, in act  
100, upon receiving the IP address list from DNS server 16, GSLB switch 12 performs, for each IP address on the  
10 IP address list (e.g., host server 26I connected to site switch 18B), a layer 4 health check and a layer 7 check. Here, layers 4 and 7 refer respectively to the transport and application protocols in the Open System Interconnection (OSI) protocol layers. The layer 4  
15 health check can be a Transmission Control Protocol (TCP) health check or a User Datagram Protocol (UDP) health check. Such a health check can be achieved, for example, by a "ping-like" operation defined under the relevant protocol. For example, under the TCP  
20 protocol, a TCP SYN packet can be sent, and the health of the target is established when a corresponding TCP ACK packet is received back from the target. In this embodiment, the layer 7 health check is provided for specified applications, such as the well-known  
25 HyperText Transport Protocol (HTTP) and the File Transfer Protocol (FTP) applications. If a host server or an associated application fails any of the health checks it is disqualified (act 102) from being the  
30 "best" site and may be excluded from the IP address list to be returned to client program 28. Since the health check indicates whether or not a host server or an associated application is available, the health check metric is suitable for use to eliminate an IP  
35 (i.e., the host server expected to provide the highest

performance). After act 100, if the list of IP addresses consists of only one IP address (act 101), the list of IP addresses is returned to client program 28 at act 108.

- 5 After act 100, if the list of candidate IP addresses for the best site consists of multiple IP addresses, it is further assessed in act 102 based upon the capacity threshold of the site switch serving that IP address. Each site switch may have a different
- 10 maximum number of TCP sessions it can serve. For example, the default number for the "ServerIron" product of Foundry Network is one million sessions, although it can be configured to a lower number. The virtual IP address configured at site switch 18B may be
- 15 disqualified from being the "best" IP address if the number of sessions for switch 18B exceed a predetermined threshold percentage (e.g., 90%) of the maximum number of sessions. (Of course, the threshold value of 90% of the maximum capacity can be changed.)
- 20 After act 102, if the list of IP addresses consists of only one IP address (act 103), the list of IP addresses is returned to client program 28 at list 108.

- After act 102, if the IP address list consists of multiple IP addresses (act 103), the remaining IP
- 25 addresses on the list can then be reordered in act 104 based upon a round-trip time (RTT) between the site switch for the IP address (e.g., site switch 18B) and the client (e.g., client 28). The RTT is computed for the interval between the time when a client machine
- 30 requests a TCP connection to a proxy server configured on a site switch, sending the proxy server a TCP SYN packet, and the time a site switch receives from the client program a TCP ACK packet. (In response to the TCP SYN packet, a host server sends a TCP SYN ACK
- 35 packet, to indicate acceptance of a TCP connection; the

009260" 48402960

client machine returns a TCP ACK packet to complete the setting up of the TCP connection.) The GSLB Switch (e.g., GSLB switch 12) maintains a database of RTT, which it creates and updates from data received

5 periodically from the site switches (e.g., site switches 18A, 18B, 22A and 22B). Each site collects and stores RTT data for each TCP connection established with a client machine. In one embodiment, the GSLB switch favors one host server over another only if the

10 difference in their RTTs with a client machine is greater than a specified percentage, the default specified percentage value being 10%. To prevent bias, the GSLB switch ignores, by default, RTT values for 5% of client queries from each responding network. After

15 act 105, if the top entries on the list of IP addresses do not have equal RTTs, the list of IP addresses is returned to client program 28 at act 108.

If multiple sites have equal RTTs then the list is reordered in act 106 based upon the location

20 (geography) of the host server. The geographic location of a server is determined according to whether the IP address is a real address or a virtual IP address ("VIP"). For a real IP address the geographical region for the host server can be

25 determined from the IP address itself. Under IANA, regional registries RIPE (Europe), APNIC (Asia/Pacific Rim) and ARIN (the Americas and Africa) are each assigned different prefix blocks. In one embodiment, an IP address administered by one of these regional

30 registries is assumed to correspond to a machine located inside the geographical area administered by the regional registry. For a VIP, the geographic region is determined from the management IP address of the corresponding site switch. Of course, a

35 geographical region can be prescribed for any IP

address to override the geographic region determined from the procedure above. The GSLB Switch prefers an IP address that is in the same geographical region as the client machine. At act 107, if the top two entries  
5 on the IP list are not equally ranked, the IP list is sent to the client program 28 at act 108.

After act 106, if multiple sites are of equal rank for the best site, the IP addresses can then be reordered based upon available session capacity (act  
10 109). For example, if switch 18A has 1,000,000 sessions available and switch 22B has 800,000 sessions available, switch 18A is then be preferred, if a tolerance limit, representing the difference in sessions available expressed as a percentage of  
15 capacity in the larger switch, is exceeded. For example, if the tolerance limit is 10%, switch 18A will have to have at a minimum 100,000 more sessions available than switch 22B to be preferred. If an IP address is preferred (act 110), the IP address will be  
20 placed at the top of the IP address list, and is then returned to the requesting entity at act 108. Otherwise, if the session capacity does not resolve the best IP address, act 111 then attempts to a resolution based upon a "flashback" speed. The flashback speed is  
25 a time required for a site switch to respond to layers 4 and 7 health checks by the GSLB switch. The flashback speed is thus a measure of the load on the host server. Again, the preferred IP address will correspond to a flashback speed exceeding the next one  
30 by a preset tolerance limit.

In one embodiment, flashback speeds are measured for well-known applications (layer 7) and their corresponding TCP ports (layer 4). For other applications, flashback speeds are measured for user  
35 selected TCP ports. Layer 7 (application-level)

flashback speeds are compared first, if applicable. If the application flashbacks fail to provide a best IP address, layer 4 flashback speeds are compared. If a host server is associated with multiple applications, the GSLB switch selects the slowest response time among the applications for the comparison. At act 112, if a best IP address is resolved, the IP address list is sent to client program 28 at act 108. Otherwise, at act 113, an IP address in the site that is least often selected to be the "best" site is chosen. The IP address list is then sent to client program 28 (act 108).

Upon receipt of the IP address list, the client's program uses the best IP address selected (i.e., the top of the list) to establish a TCP connection with a host server. Even then, if there is a sudden traffic surge that causes a host server to be overloaded, or if the host servers or the applications at the site become unavailable in the mean time, the site switch can redirect the TCP connection request to another IP address using, for example, an existing HTTP redirection procedure. The present invention does not prevent a site switch from performing load balancing among host servers within its sub-network by redirection using a similar mechanism.

To provide an RTT under the present invention described above, at the first time a client accesses an IP address, a site switch (e.g., site switch 22A of Fig. 2) monitors the RTT time -- the time difference between receiving a TCP SYN and a TCP ACK for the TCP connection -- and records it in an entry of the cache database. The RTT time measured this way corresponds to the natural traffic flow between the client machine and the host sever specified, rather than an artificial RTT based on "pinging" the client machine under a

standard network protocol. Periodically, the site switches report the RTT database to a GSLB switch along with load conditions (e.g., number of sessions available). The GSLB switch aggregates the RTTs reported into a proximity table indexed by network neighborhood. (A network neighborhood is the portion of a network sharing a prefix of an IP address.) The GSLB switch can thus look up the RTT for a client machine to any specific host server, based on the client's network neighborhood specified in the client's IP address. From the accesses to the host servers from a large number of network neighborhoods, the GSLB switch can build a comprehensive proximity knowledge database that enables smarter site selection. In order to keep the proximity table useful and up-to-date, the GSLB switch manages the proximity table with cache management policies (e.g., purging infrequently used entries in favor of recently obtained RTTs). The proximity data can be used for all IP addresses served by each site switch.

While particular embodiments of the present invention have been shown and described it will be apparent to those skilled in the art that changes and modifications may be made without departing from this invention in its broader aspect and, therefore, the appended claims are to encompass within their scope all such changes and modifications.

Claims

I claim:

1. A method of providing load balancing among host servers in a computer network using a load  
5 balancing switch and a plurality of site switches, the method comprising:

10 coupling said load balancing switch between said computer network and an authoritative domain name server and configuring said load balancing switch as a proxy to said authoritative domain name server;

coupling each of said host servers to said computer network through said site switches;

15 collecting at said load balance switch a first set of performance metrics regarding said network;

20 whenever said authoritative domain name server provides network addresses in response to a query regarding a domain name, arranging said network addresses as an ordered list in accordance with said performance metrics;

forwarding said ordered list of network addresses as a response to said query to an originator of said query.

- 25 2. A method as in Claim 1, further comprising:

collecting a second set of performance metrics regarding said network, said second set of performance metrics reflecting access conditions

to said host servers at each of said site switches;

5        sending said second set of performance metrics from said site switches to said load balancing server; and

including said second set of performance metrics with said first set of performance metrics.

3.     The method of Claim 1, wherein said first set of performance metrics includes a health check sent  
10     from said load balancing switch to each of said site switches.

4.     The method of Claim 3 wherein, when any of said host servers fails said health check, a network address of said failed host server is provided a lesser  
15     position in said ordered list.

5.     The method of Claim 2, wherein said collection of said second set of performance metrics includes recording, at each site switch, a number of sessions connected to host servers having network  
20     addresses configured on said site switch.

6.     The method of Claim 5, wherein when said number of sessions at said site switch exceeds a , predetermined percentage of that site switch's maximum capacity, a corresponding one of said network addresses  
25     is provided a lesser position in said ordered list.

7.     The method of Claim 2, wherein said collecting said second set of performance metrics

includes recording, at each site switch, a round trip time indicative of elapse time for exchanging messages between each site switch and a client machine of said computer network.

5           8.    The method of Claim 7, wherein said round trip time being an actual recorded time period between said site switch receiving a connection request from said client machine and said site switch receiving an acknowledgement of a connection from said client  
10 machine.

          9.    The method of Claim 1, wherein said arranging takes into consideration the geographical location of said originator of said query.

          10.   The method of Claim 3, wherein said  
15 collecting of said first set of performance metrics includes recording a time interval for each site switch between said load balancing switch initiating said health check and said load balancing switch receiving a response from said site switch.

20           11.   The method of Claim 1, wherein said arranging selects a network address of a least recently selected host server for placement at a higher position in said ordered list.

          12.   The method of Claim 1 further comprising said  
25 load balancing switch limiting a valid time for each network address in said ordered list to less than a predetermined value.

13. The method of Claim 1, further comprising,  
when a connection request is received at a site switch  
for a connection to one of said host servers, said site  
switch redirecting said connection request to another  
5 one of said host servers.

14. A system for balancing load among host  
servers in a computer network, comprising:

an authoritative domain name server;

10 a load balancing switch coupled to said  
authoritative domain name server, said load  
balancing switch (a) being configured to be a  
proxy to said authoritative domain name server;  
(b) collecting a first set of performance metrics  
15 regarding said network; and (c) arranging a list  
of network addresses from said authoritative  
domain name server in accordance with said first  
set of performance metrics; and

a plurality of site switches coupling said host  
servers to said computer network.

20 15. A system as in Claim 14, wherein each of said  
site switches (a) collects a second set of performance  
metrics regarding said network, said second set of  
performance metrics reflecting access conditions to  
host servers at said site switch; and (b) sends said  
25 second set of performance metrics to said load  
balancing server; whereupon said load balancing switch  
includes said second set of performance metrics with  
said first set of performance metrics.

16. A system as in Claim 14, wherein said first set of performance metrics includes a health check sent from said load balancing switch to each of said site switches.

5        17. A system as in Claim 16 wherein, when any of said host servers fails said health check, a network address of said failed host server is provided a lesser position in said ordered list.

10        18. A system as in Claim 15, wherein said second set of performance metrics includes a number of sessions connected to a network address configured at said site switch.

15        19. A system as in Claim 18, wherein when said number of sessions exceeds a predetermined percentage of that site switch's maximum capacity, said network address is provided a lesser position in said ordered list.

20        20. A system as in Claim 15, wherein said second set of performance metrics includes a round trip time indicative of elapse time for exchanging messages between each host server coupled to said site switch and a client machine of said computer network.

25        21. A system as in Claim 20, wherein said round trip time being an actual recorded time period between said site switch receiving a connection request from said client machine and said site switch receiving an acknowledgement of a connection from said client machine.

22. A system as in Claim 14, wherein said arranging takes into consideration the geographical location of said originator of said query.

23. A system as in Claim 17, wherein said first  
5 set of performance metrics includes a time interval for each site switch between said load balancing switch initiating said health check and said load balancing switch receiving a response from said site switch.

24. A system as in Claim 14, wherein said  
10 arranging selects a network address of a least recently selected host server for placement at a higher position in said ordered list.

25. A system as in Claim 14, wherein said load  
15 balancing switch limits a valid time for each network address in said ordered list to less than a predetermined value.

26. A system as in Claim 14, wherein when a  
20 connection request is received at a site switch for a connection to one of said host servers, said site switch redirects said connection request to another one of said host servers.

# GLOBAL SERVER LOAD BALANCING

Ivy Pei-Shan Hsu

David Chun Ying Cheung

Rajkumar Ramniranjan Jalan

5

## Abstract of The Disclosure

10 A global server load balancing (GSLB) switch  
serves as a proxy to an authoritative DNS communicates  
with numerous site switches which are coupled to host  
servers serving specific applications. The GSLB switch  
receives from site switches operational information  
regarding host servers within the site switches  
neighborhood. When a client program requests a  
resolution of a host name, the GSLB switch, acting as a  
15 proxy of an authoritative DNS, returns one or more  
ordered IP addresses for the host name. The IP  
addresses are ordered using metrics that include the  
information collected from the site switches. In one  
instance, the GSLB switch places the address that is  
20 deemed "best" at the top of the list.

The diagram illustrates a cloud-based system architecture. A central cloud, labeled 14, is connected to several components:

- A database, labeled 16, is connected to the cloud via a network interface, labeled 12.
- A server rack, labeled 20, is connected to the cloud via a network interface, labeled 19. This rack contains multiple server units, labeled 26A, 26B, 26E, and 26I, which are connected to a central switch, labeled 18A and 18B.
- A server rack, labeled 24, is connected to the cloud via a network interface, labeled 21. This rack contains multiple server units, labeled 26K, 26M, 26L, and 26N, which are connected to a central switch, labeled 22A and 22B.
- A mobile device, labeled 28, is connected to the cloud via a wireless connection, labeled 30.

Fig. 1

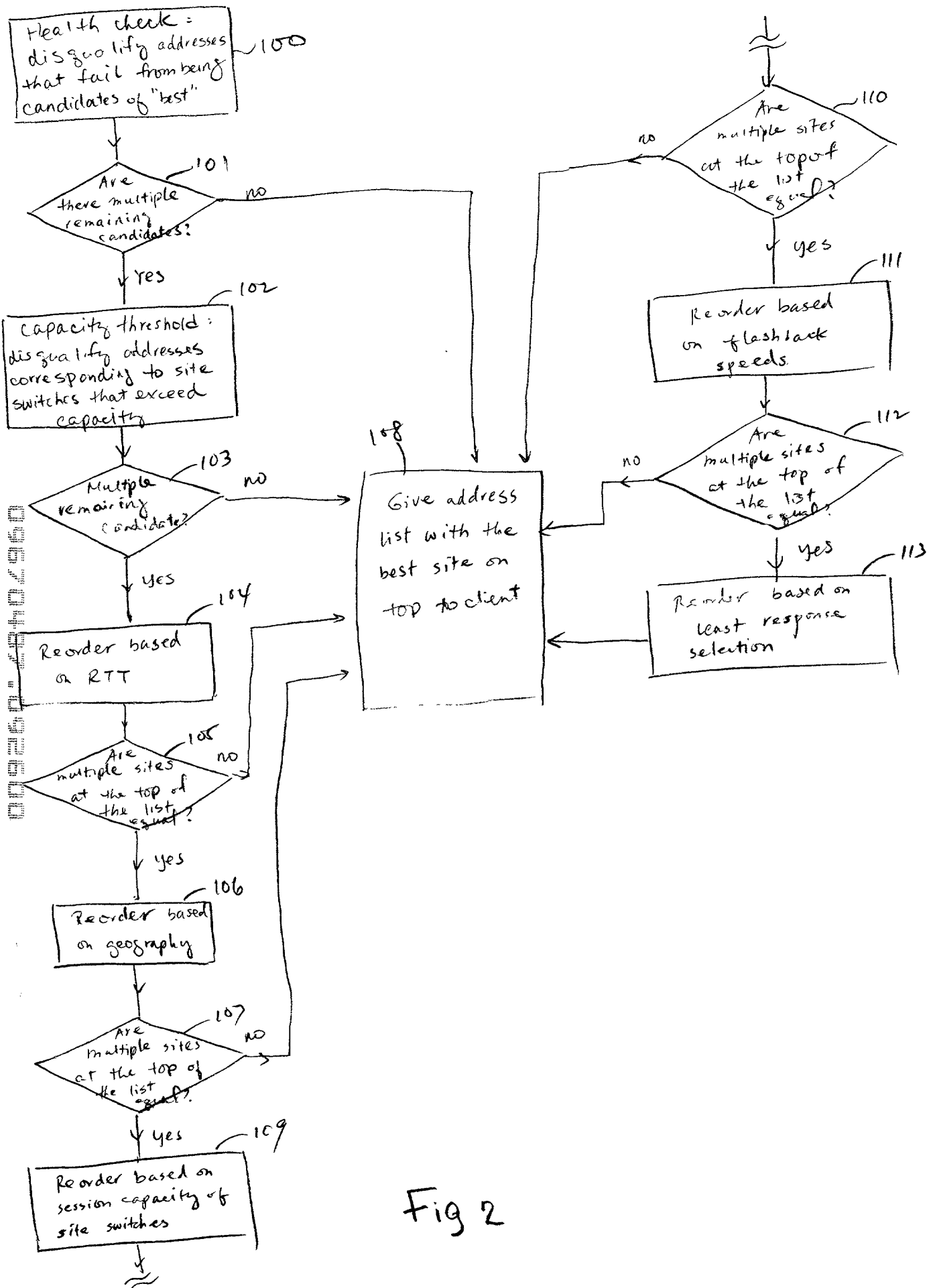


Fig 2

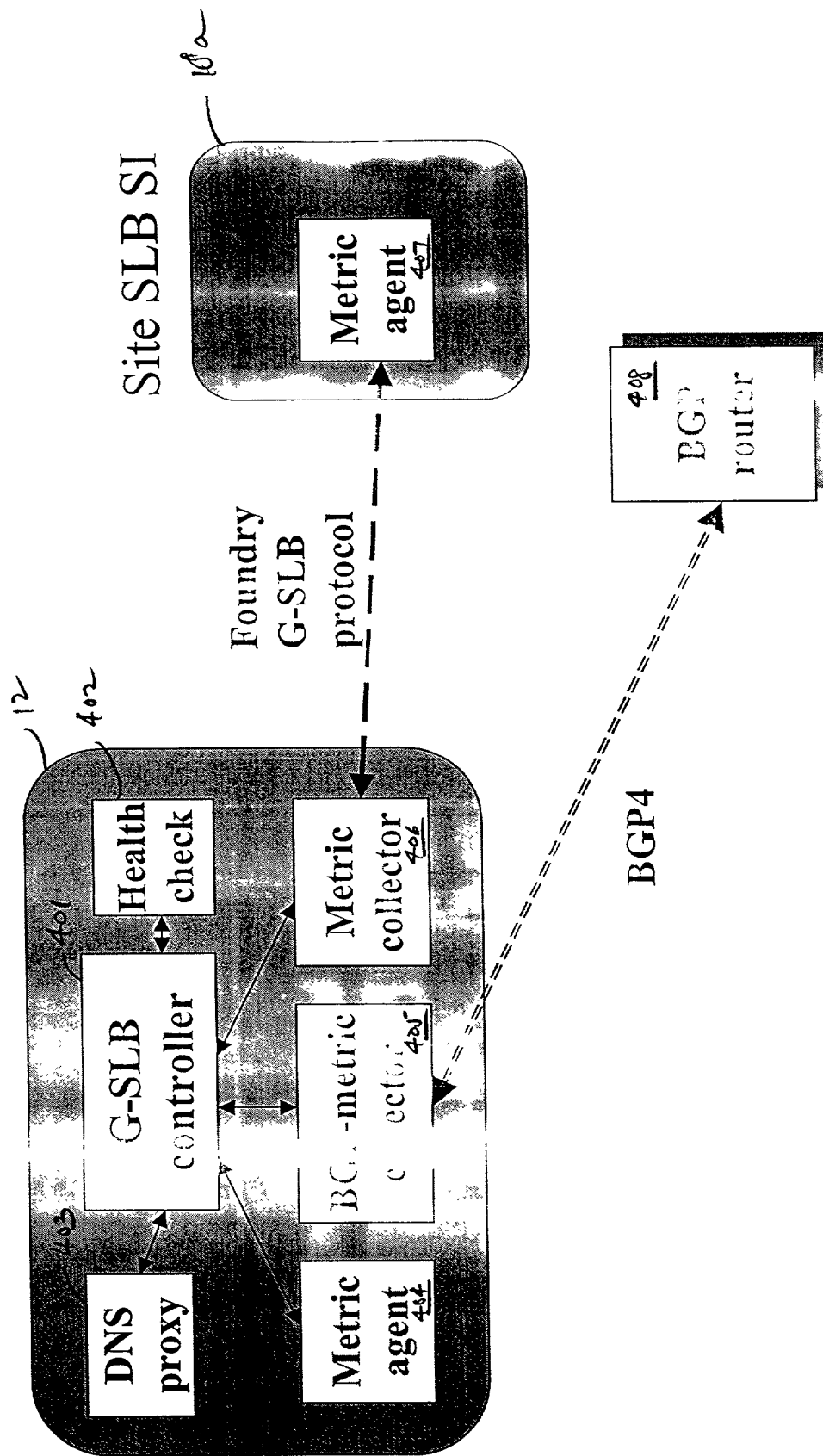


FIG. 3

# DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below adjacent to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of subject matter (process, machine, manufacture, or composition of matter, or an improvement thereof) which is claimed and for which a patent is sought by way of the application entitled

## "Global Server Load Balancing"

which (check) ☒ is attached hereto.  
☐ and is amended by the Preliminary Amendment attached hereto.  
☐ was filed on      as Application Serial No.  
☐ and was amended on      (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information, which is material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, § 119(a)-(d) of any foreign application(s) for patent or inventor's certificate or any PCT international application(s) designating at least one country other than the United States of America listed below and have also identified below any foreign application(s) for patent or inventor's certificate or any PCT international application(s) designating at least one country other than the United States of America filed by me on the same subject matter having a filing date before that of the application(s) of which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
Number	Country	Day/Month/Year Filed	Yes	No
N/A	{Country}		<input type="checkbox"/>	<input type="checkbox"/>

I hereby claim the benefit under Title 35, United States Code, § 119(e) of any United States provisional application(s) listed below:

Provisional Application Number	Filing Date
N/A	

I hereby claim the benefit under Title 35, United States Code, § 120 of any United States application(s) or PCT international application(s) designating the United States of America listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior application(s) in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose information, which is material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56, which became available between the filing date of the prior application(s) and the national or PCT international filing date of this application:

Application Serial No.	Filing Date	Status (patented, pending, abandoned)
N/A		{Status}

009260 48402960

I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and to transact all business in the United States Patent and Trademark Office connected therewith:

Alan H. MacPherson (24,423); Brian D. Ogonowsky (31,988); David W. Heid (25,875); Norman R. Klivans (33,003); Edward C. Kwok (33,938); David E. Steuber (25,557); Michael Shenker (34,250); Stephen A. Terrile (32,946); Peter H. Kang (40,350); Ronald J. Meetin (29,089); Ken John Koestner (33,004); Omkar K. Suryadevara (36,320); David T. Millers (37,396); Michael P. Adams (34,763); Robert B. Morrill (43,817); James E. Parsons (34,691); Philip W. Woo (39,880); Emily Haliday (38,903); Tom Hunter (38,498); Michael J. Halbert (40,633); Gary J. Edwards (41,008); Daniel P. Stewart (41,332); John T. Winburn (26,822); Tom Chen (42,406); Fabio E. Marino (43,339); Don C. Lawrence (31,975); Marc R. Ascolese (42,268); Carmen C. Cook (42,433); David G. Dolezal (41,711); Roberta P. Saxon (43,087); Mary Jo Bertani (42,321); Dale R. Cook (42,434); Sam G. Campbell (42,381); Matthew J. Brigham (44,047); Patrick D. Benedicto (40,909); T.J. Singh (39,535); Shireen Irani Bacon (40,494); Rory G. Bens (44,028); George Wolken, Jr. (30,441); John A. Odozynski (28,769); Paul E. Lewkowicz (44,870); Theodore P. Lopez (44,881); Eric Stephenson (38,321); Christopher Allenby (45,906); David C. Hsia (46,235); Mark J. Rozman (42,117); Margaret M. Kelton (44,182); Do Te Kim (46,231); Alex Chen (45,591); Monique M. Heyninck (44,763); Gregory J. Michelson (44,940); Jonathan Geld (44,702); Emmanuel Rivera (45,760); Jason FarHadian (42,523); Matthew J. Spark (43,453); and Elaine H. Lo (41,158).

Please address all correspondence and telephone calls to:

Edward C. Kwok  
**SKJERVEN MORRILL MacPHERSON LLP**  
 25 Metro Drive, Suite 700  
 San Jose, California 95110-1349

Telephone: 408-453-9200  
 Facsimile: 408-453-7979

I declare that all statements made herein of my own knowledge are true, all statements made herein on information and belief are believed to be true, and all statements made herein are made with the knowledge that whoever, in any matter within the jurisdiction of the Patent and Trademark Office, knowingly and willfully falsifies, conceals, or covers up by any trick, scheme, or device a material fact, or makes any false, fictitious or fraudulent statements or representations, or makes or uses any false writing or document knowing the same to contain any false, fictitious or fraudulent statement or entry, shall be subject to the penalties including fine or imprisonment or both as set forth under 18 U.S.C. 1001, and that violations of this paragraph may jeopardize the validity of the application or this document, or the validity or enforceability of any patent, trademark registration, or certificate resulting therefrom.

009260 42404960

1.

Full name of first joint inventor:

Ivy Pei-Shan Hsu

Inventor's Signature: Ivy Pei-Shan Hsu

Date:

9/7/2000

Residence:

Danville, California 94506

Post Office Address:

214 Stetson Drive  
Danville, California 94506

Citizenship:

Taiwan

2.

Full name of joint inventor:

David Chun Ying Cheung

Inventor's Signature: David Cheung

Date:

9/20/2000

Residence:

San Jose, California 95134

Post Office Address:

403-13 Camille Circle  
Danville, California 95134

Citizenship:

United States

3.

Full name of joint inventor:

Rajkumar Ramniranjan Jalan

Inventor's Signature: Rajkumar Jalan

Date:

09/20/2000

Residence:

Fremont, California 94555

Post Office Address:

33685 Simple Court  
Fremont, California 94555

Citizenship:

India

09/20/2000 13:49:36